CLAIMS

What is claimed is:

1. A system of memory management for persistent storage over a reboot of a set of data from an application executing on a circuit card comprising:

a memory for persistent storage of data over a reboot, the memory being located on the circuit card; and

a memory manager for directly controlling access to the memory, the memory manager executing on the circuit card.

2. The system of claim 1 wherein the memory manager has a message handler for receiving a request for storage of the set of data from the application wherein the request includes an application identifier.

3. The system of claim 2 wherein the request includes the set of data from the application.

4. The system of claim 2 wherein the request includes a reboot state indicator indicating a state of a reboot during which the set of data is to be sent to the application.

5. The system of claim 2 wherein the memory has a first memory region marked active and a second memory region marked alternate and the memory manager performs the following acts responsive to receiving the request for storage of the set of data from the application:

writing the set of data and its application identifier to the second memory region marked alternate;

marking the second memory region marked alternate as being marked active;

marking the first memory region marked active as being marked alternate; and

copying data from the second memory region marked active to the first memory region marked alternate.

6. The memory manager of claim 4 further performs:

storing the reboot state identifier for the set of data in the memory.

7. The system of claim 6 wherein the memory has an active memory region and an alternate memory region and the memory manager performs the following responsive to a reboot of the card occurring:

performing an integrity test for the active memory region; and

responsive to a successful integrity test result indicating valid data for the active region, providing the set of data from the active region to the application during the state of the reboot indicated by the stored reboot state indicator for the set.

8. The memory manager module of claim 7 further performing:

responsive to an unsuccessful integrity test result for the active region, performing an integrity test for the alternate memory region; and

responsive to a successful integrity test result for the alternate region, providing the set of data from the alternate region to the application during the state of the reboot indicated by the stored reboot state indicator for the set, the set of data.

9. The system of claim 6 wherein the memory comprises a cold-reboot persistence memory.

10. The system of claim 9 wherein the memory manager comprises a cold-reboot module for performing the following:

receiving a commitment indicator indicating that the set of data is to be saved to cold-reboot persistence memory; and

writing the set of data to the cold- reboot persistence memory.

11. The system of claim 1 wherein the memory is a memory having an active memory region and an alternate memory region and the memory manager comprises a data size change module for performing the following for adjusting for a new version of a set of data:

overwriting the alternate memory region with data from the active memory region up to a location in the alternate memory region associated with a stored version of the set of data;

at the location associated with the stored version of the set of data, writing the new version of the set of data in the alternate region;

writing the data from a location in the active region located after a stored version set of data to the alternate memory region starting in a location after where the new version of the set of data has been stored; and

marking the alternate memory region as the active memory region.

12. The circuit card of claim 1 wherein the circuit card is a node element module card having an optical element.

13. A system of memory management for persistent storage over a reboot of a set of data from an application executable on a first circuit card in a memory located on a second circuit card comprising:

the first circuit card having a first node element module for performing a payload traffic-carrying function and a first copy of a software application;

the second circuit card having a second node element module for performing a payload traffic-carrying function and a second copy of the same application;

each card comprising a memory management system comprising a memory for persistent storage of data over a reboot, and a memory manager for directly controlling access to the memory;

the memory manager of the second card having a message handler for receiving a request from the first card for storage of the set of data from the first copy of the application in the memory of the second card, the request including an application identifier for the set of data; and

the memory manager of the second card comprising instructions for storing the set of data and the application identifier in the memory of the second card responsive to the request.

14. The system of claim 13 wherein the application identifier comprises an endpoint identification identifying the second card as a destination for the set of data.

15. The system of claim 13 wherein the application identifier further comprises an object identifier identifying an object in the first copy of the application with which the set of data is associated.

16. A method of memory management for persistent storage of a set of data from an application executing on a circuit card comprising:

receiving from the application a request for storage of the set of data in a memory for persistent storage of data over a reboot, the memory being located on the circuit card, wherein the request includes an application identifier; and

directly controlling access to the memory by a memory manager executing on the circuit card.

17. The method of claim 16 further comprising storing a reboot state indicator for the set of data in the memory, the reboot state indicator indicating a state of a reboot during which the set of data is to be sent to the application.

18. The method of claim 17 wherein the memory has an active memory region and an alternate memory region and wherein directly controlling access to the memory by the memory manager comprises:

writing the set of data and its application identifier to the alternate memory region;

performing an integrity test on the alternate memory region; and

responsive to a successful integrity test result indicating valid data, marking the active memory region as the alternate region, and marking the alternate region as the active region.

19. The method of claim 17 wherein the memory has an active memory region and an alternate memory region, the method further comprising performing the following responsive to a reboot of the card occurring:

performing an integrity test for the active memory region; and

responsive to a successful integrity test result for the active region, providing each set of data from the active region to the application during the state of the reboot indicated by the stored reboot state indicator for the set.

20. The method of claim 19 further performing:

responsive to an unsuccessful integrity test result for the active region, performing an integrity test for the alternate memory region; and

responsive to a successful integrity test result for the alternate region, providing the set of data from the alternate region to the application during the state of the reboot indicated by the stored reboot state indicator for the set.

21. The method of claim 17 wherein the memory comprises a cold-reboot persistence memory.

22. The method of claim 21 further comprising:

receiving a commitment indicator indicating that the set of data is to be saved to cold-reboot persistence memory; and

writing the set of data to the cold- reboot persistence memory.

23. The method of claim 16 wherein the memory is a memory having an active memory region and an alternate memory region and the method further comprises performing the following for adjusting for a new version of a set of data:

overwriting the alternate memory region with data from the active memory region up to a location in the alternate memory region associated with a stored version of the set of data;

at the location associated with the stored version of the set of data, writing the new version of the set of data in the alternate region;

writing the data from a location in the active region located after a stored version of the set of data to the alternate memory region starting in a location after where the new version of the set of data has been stored; and

marking the alternate memory region as the active memory region.

24. The method of claim 16 where in the circuit card is a node element module circuit card having an optical element.

25. In a system of memory management for persistent storage over a reboot of a set of data from an application executable on a first circuit card in a memory located on a second circuit card wherein the first circuit card has a first node element module having an optical element and a first copy of a software application, the second circuit card has a second node element module having an optical element and a second copy of the same application, and each card comprises a memory management system comprising a memory for persistent storage of data over a reboot, and a memory manager for directly controlling access to the memory, a method of memory management for persistent storage over a reboot of a set of data from an application executing on a first circuit card in a memory located on a second circuit card comprising:

receiving a request by the memory manager of the second card from the first card for storage of the set of data from the first copy of the application in the memory of the second card, the request including an application identifier for the set of data; and

storing of the set of data and the application identifier in the memory of the second card responsive to the request.

26. The method of claim 25 wherein the application identifier comprises an endpoint identification identifying the second card as a destination for the set of data.

27. The method of claim 25 wherein the application identifier further comprises an object identifier identifying an object in the first copy of the application with which the set of data is associated.

28. A computer usable medium comprising instructions embodied thereon, which when executed by a processor cause the processor to perform a method of memory management

for persistent storage of a set of data from an application executing on a circuit card, the method comprising:

receiving from the application a request for storage of the set of data in a memory for persistent storage over a reboot, the memory being located on the circuit card, wherein the request includes an application identifier; and

directly controlling access to the memory by a memory manager executing on the circuit card.

29. A system of memory management for persistent storage over a reboot of a set of data from an application executing on a circuit card comprising:

storage means for persistent storage of data over a reboot, the storage means being located on the circuit card; and

means for directly controlling access to the storage means, the means for directly controlling access executing on the circuit card.